1)

Requirements:

- Software allows employees requests an elevator to go to a specific floor.

- Not all elevators visit all floors, more specifically only Elevator # 3 can visit the "VIP Suite" (implemented as Floor #4).

- Access to VIP Suite is restricted by employee rank.

- Software monitors elevators status on the elevator panel, installed on the first floor.

- Elevator control system is designed to be as efficient as possible, to avoid congestions.

Superficial:

No superficial in my opinion.

Improvements:

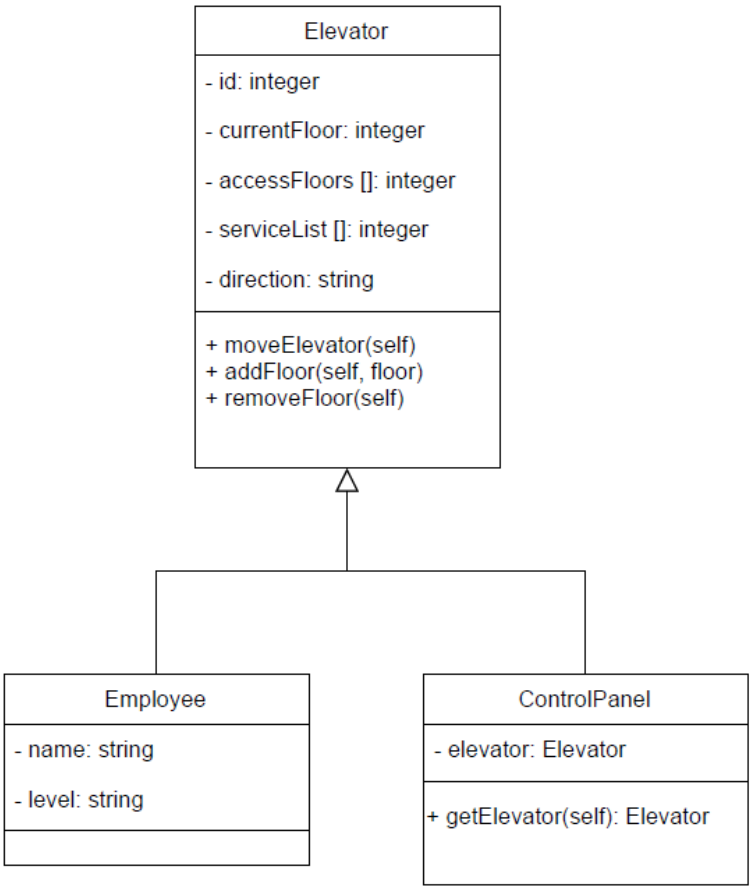
- Implementing a "help" button that connects with the building security system or 911, if building security system doesn't answer.

- Implementing a "fire" button that moves the elevators to safe floors, plays an audio to alert people inside the elevator and display a text message on screen.

- Implementing voice over and having bump dot buttons for visually impaired and deaf people.

- Including a weight sensor to detect if elevator's total weight is over the safe limit and play a message to ask Employees to offload from the elevator until it reaches a safe weight limit.

2)



Elevator: An elevator is a class that has ID, currentFloor, accessFloors, serviceList and direction. serviceList represents the floors that are selected, accessFloors are the floors accessible by the elevator.

ControlPanel: A control panel is a class that has an elevator assigned to it. It only calls an elevator using the getElevator() function that uses a helper function called, computeClosestElevator().

Employee: An employee is a class that has name and level. Level represents their rank in company and it makes a difference in the floors they can visit.

Code base is implemented using object-oriented architecture. Main concepts used utilized are:

1) Testable 4) Robustness

2) Extensible 5) Maintainable

3) Reusable 6) Understandable